DroMOOC

Computer Vision

Basic Level

Feature Matching and Optical Flow

Antoine Manzanera

ENSTA

Feature Matching and Optical Flow

Objectives of the Lecture

This lecture is about motion estimation from video. It presents the underlying theory of optical flow and its interest for vision based navigation of autonomous drones. It also presents the main algorithms for computing the optical flow in practice.

Outline of the Lecture

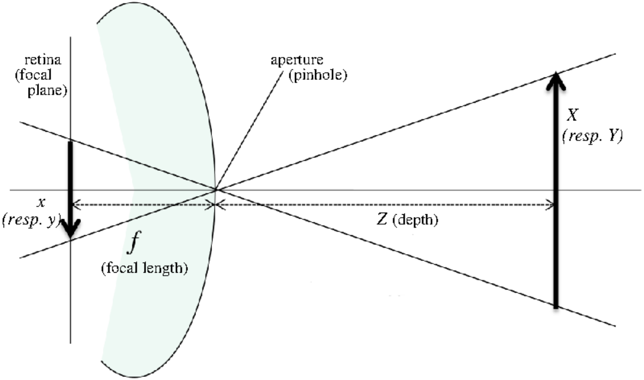
- What is the optical flow?
- What is it used for?
- Sparse (local) estimation of the optical flow
- Dense (global) estimation of the optical flow
- Multiscale approaches

Optical Flow: definition

- The *theoretical optical flow* is the 2d vector field of the apparent velocity of pixels.
- It corresponds to the projection of the 3d velocity of points projected onto the focal plane, with respect to the camera coordinate system.
- The *practical optical flow* is a collection of 2d vectors of apparent velocity estimated from two (or more) consecutive images.

Theoretical Optical Flow

The theoretical optical flow can be derived from the equations of projection of a 3d point (X, Y, Z) onto the focal plane (pinhole model): $x = \frac{fX}{Z}$ and $y = \frac{fY}{Z}$:



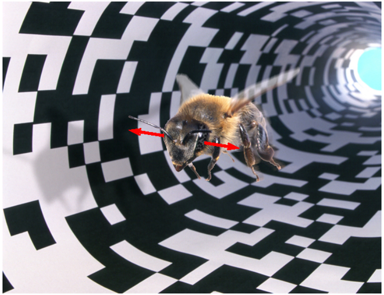
Theoretical Optical Flow

- $\dot{x} = f\left(\frac{\dot{X}}{Z} - \frac{X\dot{Z}}{Z^2}\right)$
- $\dot{y} = f\left(\frac{\dot{Y}}{Z} - \frac{Y\dot{Z}}{Z^2}\right)$

(Notation: $\dot{m} = \frac{\partial m}{\partial t}$)

Particular Optical Flow: Horizontal travelling

If moving into a static scene, the 3d velocity vector $(\dot{X}, \dot{Y}, \dot{Z})$ is the same everywhere! In the particular case of a camera moving laterally, i.e. such that $\dot{Z} = 0$ (e.g. pure translation along OX axis), we get:



From [Tautz 08]

Horizontal travelling

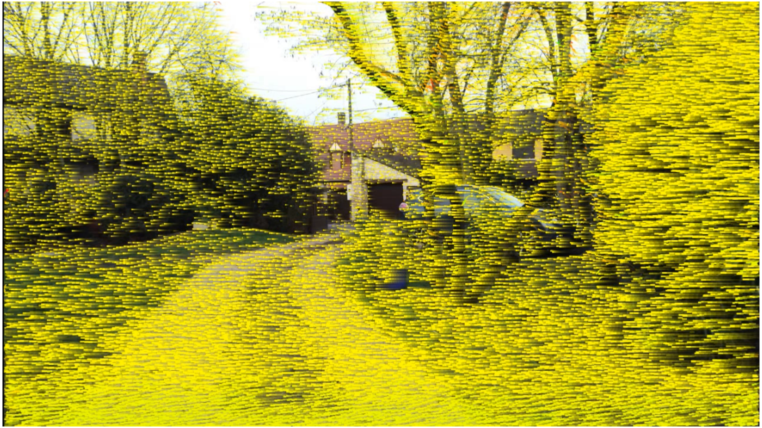
$$\dot{x} = f\frac{\dot{X}}{Z} \iff Z = \frac{f\dot{X}}{\dot{x}}$$

The depth of a point is inversely proportional to its apparent speed.

Applications:

- Centering behaviour with two laterally viewing cameras.

Example on a computed flow on a drone fly



Horizontal (Right) travelling

Drone: Parrot Anafi

Pilot: C. Pinard

Optical flow: **[Garrigues 14]**

Example on a computed flow on a drone fly

Vertical (Up) travelling

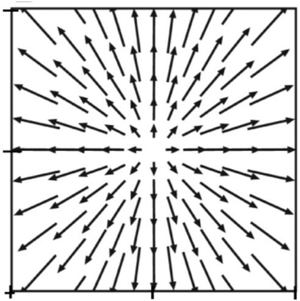
Drone: Parrot Anafi

Pilot: C. Pinard

Optical flow: **[Garrigues 14]**

Particular Optical Flow: Radial zoom

In the particular case of a camera moving in the direction of its optical axis, (pure translation along OZ axis), we get:



Radial zoom

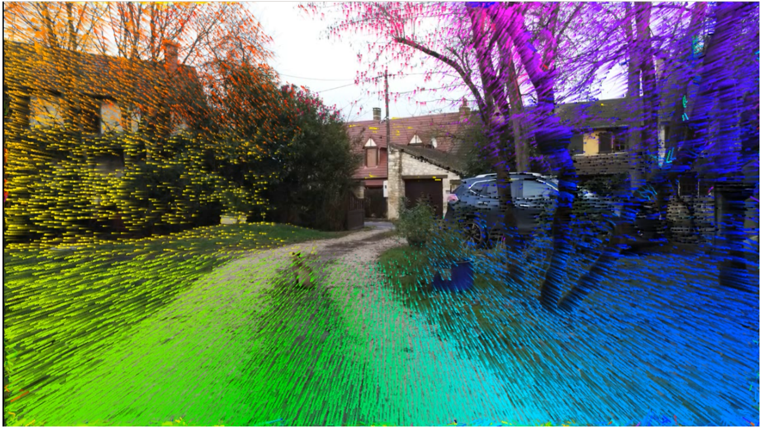
- $\dot{x} = -f\frac{X\dot{Z}}{Z^2} = -x\frac{\dot{Z}}{Z}$
- $\dot{y} = -f\frac{Y\dot{Z}}{Z^2} = -y\frac{\dot{Z}}{Z}$

The optical flow vectors di(con)verge from (toward) the direction of motion (Focus of Expansion).

Applications:

- Automatic landing
- Vision based stabilisation

Example on a computed flow on a drone fly



Forward zoom

Drone: Parrot Anafi

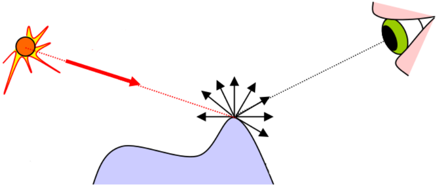
Pilot: C. Pinard

Optical flow: **[Garrigues 14]**

Difficulties and Constraints of the Practical Optical Flow

The practical optical flow is estimated from consecutive images, and is based on the assumption of *appearance consistency over time*. It is then based on the following requisites and constraints:

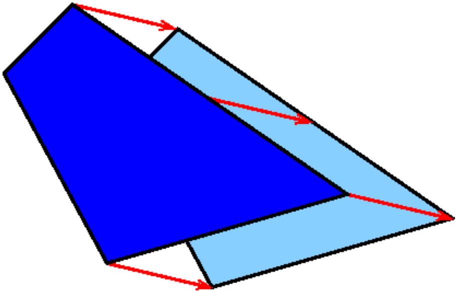
- *Lambertian reflection*
- *Illumination consistency*
- *Unambiguous Texturing*



Difficulties and Constraints of the Practical Optical Flow

Another difficulty is the *Aperture Problem*: in absence of sufficient structure...

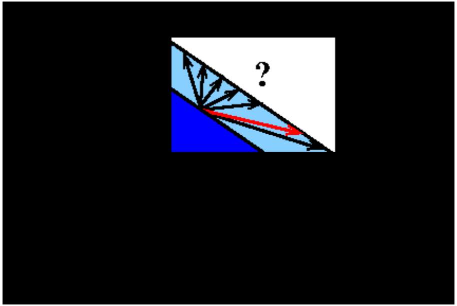
- *Illumination consistency*
- *Lambertian reflection*
- *Unambiguous Texturing*



Difficulties and Constraints of the Practical Optical Flow

...the motion can be at best estimated along the direction of the spatial gradient.

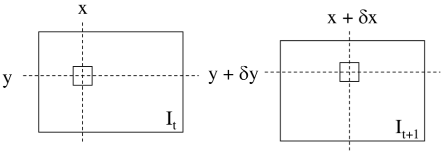
- *Illumination consistency*
- *Lambertian reflection*
- *Unambiguous Texturing*

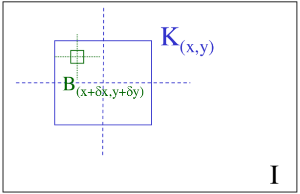


Local matching measures for Optical Flow estimation

The basic method for estimating the velocity (v_x^t, v_y^t) at pixel (x, y) and time t is minimising a local matching (sum-of-squared) distance like:

$$M^t_{(x,y)}(\delta x, \delta y) = \sum_{(b_1,b_2)\in B} \left(I(x+b_1, y+b_2, t) - I(x+\delta_x+b_1, y+\delta y+b_2, t+\delta t)\right)^2$$





$$(v_x^t, v_y^t) = \arg \min_{(\delta x,\delta y)\in K} M^t_{(x,y)}(\delta x, \delta y)$$

- B is a small neighbourhood (patch).
- K is the search domain for displacements.

Local Lucas and Kanade's Optical Flow method

Lucas and Kanade's method **[Lucas & Kanade 81]** is based on:

1. Approximating the Matching measure $M^t_{(x,y)}(\delta x, \delta y)$ by assuming the image I *regular* and the displacement $(\delta x, \delta y)$ *small*.
2. Finding the minimum of the approximated function as the point where its derivatives with respect to δx and δy are equal to zero.

Local Lucas and Kanade's Optical Flow method

Lucas and Kanade's method **[Lucas & Kanade 81]** is based on:

1. Approximating the Matching measure $M^t_{(x,y)}(\delta x, \delta y)$ by assuming the image I *regular* and the displacement $(\delta x, \delta y)$ *small*.
2. Finding the minimum of the approximated function as the point where its derivatives with respect to δx and δy are equal to zero.

$$I(x + \delta_x, y + \delta y, t + \delta t) \simeq I(x, y, t) + \tfrac{\partial I}{\partial x}\delta x + \tfrac{\partial I}{\partial y}\delta y + \tfrac{\partial I}{\partial t}\delta t$$

$$M^t_{(x,y)}(\delta x, \delta y) \simeq \sum_{(x,y)\in B} \left(\frac{\partial I}{\partial x}\delta x + \frac{\partial I}{\partial y}\delta y + \frac{\partial I}{\partial t} \right)^2$$

$$(\delta t = 1)$$

Local Lucas and Kanade's Optical Flow method

Lucas and Kanade's method **[Lucas & Kanade 81]** is based on:

1. Approximating the Matching measure $M^t_{(x,y)}(\delta x, \delta y)$ by assuming the image I *regular* and the displacement $(\delta x, \delta y)$ *small*.
2. Finding the minimum of the approximated function as the point where its derivatives with respect to δx and δy are equal to zero.

$$\arg\min_{(\delta x, \delta y)} \sum_{(x,y)\in B} \left(\frac{\partial I}{\partial x}\delta x + \frac{\partial I}{\partial y}\delta y + \frac{\partial I}{\partial t} \right)^2$$

$$2 \sum_{(x,y)\in B} \left(\frac{\partial I}{\partial x}\delta x + \frac{\partial I}{\partial y}\delta y + \frac{\partial I}{\partial t} \right) \frac{\partial I}{\partial x} = 0$$

$$2 \sum_{(x,y)\in B} \left(\frac{\partial I}{\partial x}\delta x + \frac{\partial I}{\partial y}\delta y + \frac{\partial I}{\partial t} \right) \frac{\partial I}{\partial y} = 0$$

Local Lucas and Kanade's Optical Flow method

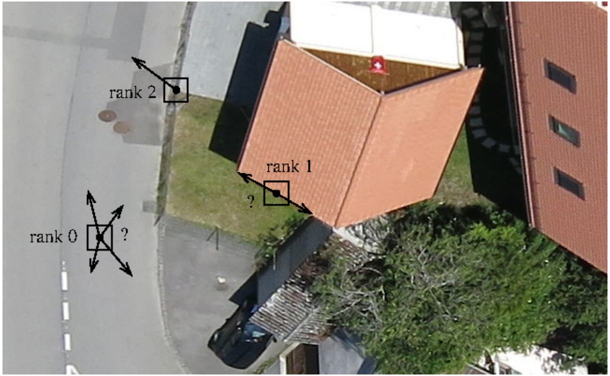
Finally the method consists in locally solving for each pixel (x, y, t) the following system:

$$\underbrace{\begin{pmatrix} \sum\limits_{(x,y)\in B} \left(\frac{\partial I}{\partial x}\right)^2 & \sum\limits_{(x,y)\in B} \frac{\partial I}{\partial x}\frac{\partial I}{\partial y} \\ \sum\limits_{(x,y)\in B} \frac{\partial I}{\partial x}\frac{\partial I}{\partial y} & \sum\limits_{(x,y)\in B} \left(\frac{\partial I}{\partial y}\right)^2 \end{pmatrix}}_{\Xi(x,y)} \cdot \begin{pmatrix} v_x^t \\ v_y^t \end{pmatrix} = \begin{pmatrix} -\sum\limits_{(x,y)\in B} \frac{\partial I}{\partial x}\frac{\partial I}{\partial t} \\ -\sum\limits_{(x,y)\in B} \frac{\partial I}{\partial y}\frac{\partial I}{\partial t} \end{pmatrix}$$

- $(v_x^t, v_y^t)^T$ is the unknown velocity vector.
- $\Xi(x, y)$ is identical to the structure matrix used in corner point detection (see Basic Course 2)!
- The system is practically solved using an iterative (Gauss-Seidel) method.

Local Lucas and Kanade's Optical Flow method

In practice LK's local method only produces reliable matching when the structure matrix Ξ has rank 2, i.e. for corner points!



Local Lucas and Kanade's Optical Flow method

The computation for each pixel being independent, Lucas and Kanade's method can be massively parallelised.
However because of its limitation related to the structure matrix, it can only provide a sparse optical flow:

LK tracker (pyramid version) from *Open CV*

Global Dense Horn and Schunck's Optical Flow method

Horn and Schunck's method **[Horn & Schunck 81]** provides a dense optical flow by minimising a global function combining:

1. A data term relating the velocity vector, the spatial gradient and the temporal gradient.
2. A regularisation term corresponding to the spatial gradient of the velocity field.

Global Dense Horn and Schunck's Optical Flow method

Horn and Schunck's method **[Horn & Schunck 81]** provides a dense optical flow by minimising a global function combining:

1. A data term relating the velocity vector, the spatial gradient and the temporal gradient.
2. A regularisation term corresponding to the spatial gradient of the velocity field.

The appearance consistency constraints writes: $I(x, y, t) = I(x + \delta x, y + \delta y, t + \delta t)$.
The first order Taylor expansion provides: $I(x, y, t) \simeq I(x, y, t) + \frac{\partial I}{\partial x}\delta x + \frac{\partial I}{\partial y}\delta y + \frac{\partial I}{\partial t}\delta t$, and then:

Optical Flow Equation

$$\nabla I.\mathbf{v} + \frac{\partial I}{\partial t} = 0$$

- $\mathbf{v} = (v_x^t, v_y^t)$ the unknown velocity vector.
- ∇I the gradient vector.
- Scalar equation with two unknowns!

Global Dense Horn and Schunck's Optical Flow method

Horn and Schunck's method provides a dense optical flow by minimising a global function combining:

1. A data term relating the velocity vector, the spatial gradient and the temporal gradient.
2. A regularisation term corresponding to the spatial gradient of the velocity field.

Horn & Schunck cost function

$$C^t_{(x,y)}(\mathbf{v}) = \left(\nabla I.\mathbf{v} + \frac{\partial I}{\partial t}\right)^2 + \lambda\left(\left(\frac{\partial v^t_x}{\partial x}\right)^2 + \left(\frac{\partial v^t_x}{\partial y}\right)^2 + \left(\frac{\partial v^t_y}{\partial x}\right)^2 + \left(\frac{\partial v^t_y}{\partial y}\right)^2\right)$$

- the 1st term is the data term.
- the 2nd term is the regularisation term.
- λ is a weighting factor.

Global Dense Horn and Schunck's Optical Flow method

$C^t_{(x,y)}(\mathbf{v})$ is finally minimised by finding the point where its derivatives with respect to v_x and v_y are equal to zero (superscript t is discarded), which leads to a system of two equations, that is solved iteratively (Gauss-Seidel), leading to:

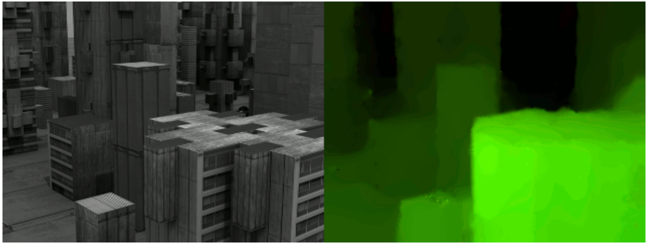
Horn & Schunck resolution algorithm

- Initialisation: $v_x^{(0)} = v_y^{(0)} = 0$.
- Repeat until convergence:
  - $v_x^{(k)} = \overline{v_x^{(k-1)}} - \frac{\partial I}{\partial x}\frac{N}{D}$
  - $v_y^{(k)} = \overline{v_y^{(k-1)}} - \frac{\partial I}{\partial y}\frac{N}{D}$

- $\overline{v}$ is the average value of v over a certain neighbourhood.
- $N = \frac{\partial I}{\partial x}\overline{v_x^{(k-1)}} + \frac{\partial I}{\partial y}\overline{v_y^{(k-1)}} + \frac{\partial I}{\partial t}$.
- $D = \lambda + \left(\frac{\partial I}{\partial x}\right)^2 + \left(\frac{\partial I}{\partial y}\right)^2$.

Global Dense Horn and Schunck's Optical Flow method

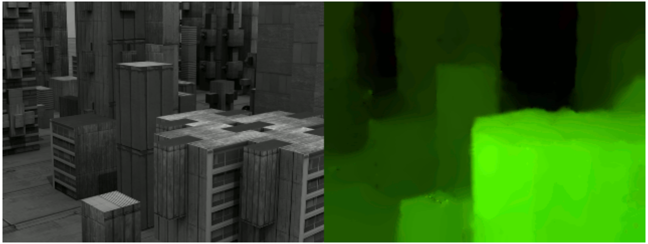
Because of the iterative estimation of the averaged (smoothed) version of $\overline{v_x}$ and $\overline{v_y}$, the Horn and Schunck's method is a global one, that cannot be parallelised the same way as Lucas and Kanade's.
But thanks to its spatial regularity hypothesis, it can provide a dense optical flow field:



Images from *boofcv.org*

Global Dense Horn and Schunck's Optical Flow method

Because of the iterative estimation of the averaged (smoothed) version of $\overline{v_x}$ and $\overline{v_y}$, the Horn and Schunck's method is a global one, that cannot be parallelised the same way as Lucas and Kanade's.
But thanks to its spatial regularity hypothesis, it can provide a dense optical flow field:



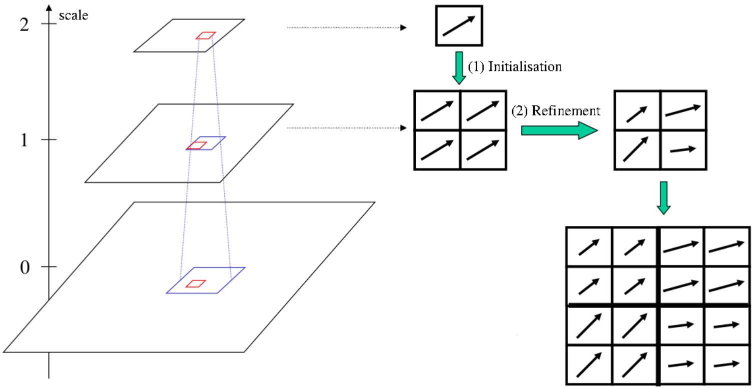
Images from *boofcv.org*

Multi-scale Optical Flow estimation

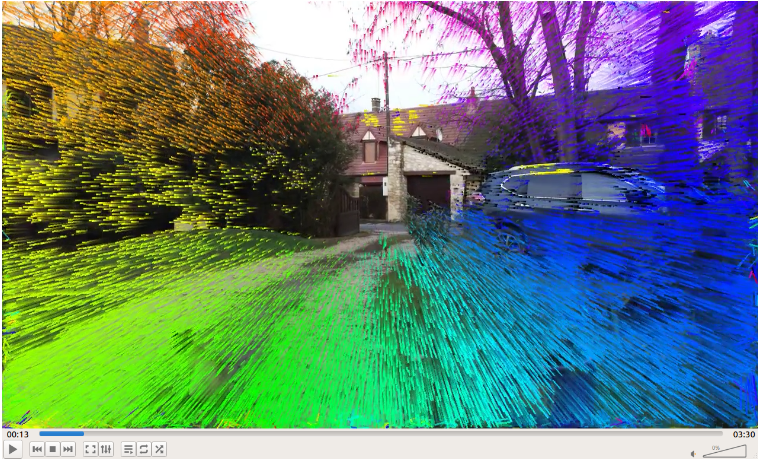
Both previous methods are strongly limited by (1) the regularity assumptions and (2) the hypothesis of small displacements.
The multi-scale optical flow estimation can be applied to any (iterative!) method by applying the estimation at different resolutions, in a coarse-to-fine manner.
The advantages are multiple:

- Estimate larger displacements.
- Mitigate the texture ambiguity.
- Densify the flow field.
- Decrease the weight of spatial regularisation.

Multi-scale Optical Flow estimation



Multi-scale Optical Flow estimation

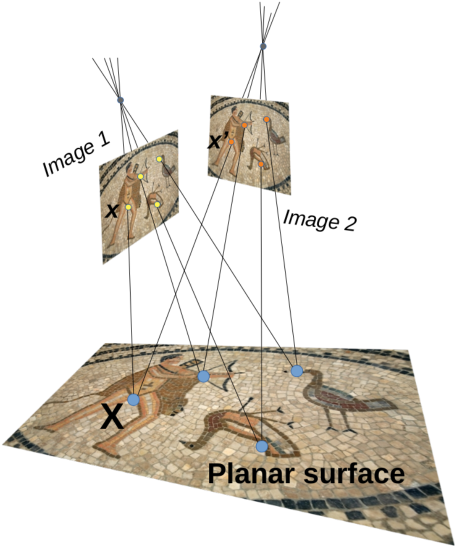


Real-time semi-dense long-term Optical Flow **[Garrigues 14]** on a Anafi fly

Drone: Parrot Anafi

Pilot: C. Pinard

One use of the optical flow: Homography estimation



- A homography is the transformation that relates two different views of the same planar surface : $x' = Hx$.
- It can be expressed, using homogeneous 2d coordinates, by a 3×3 matrix with 8 degrees of freedom.
- At least 4 pairs of matching points are necessary to estimate a homography.

One use of the optical flow: Homography estimation

- $x' = Hx$ in 2d homogeneous coordinates, means:

$$\lambda \begin{pmatrix} x'_1 \\ x'_2 \\ 1 \end{pmatrix} = \begin{pmatrix} H_1^1 & H_1^2 & H_1^3 \\ H_2^1 & H_2^2 & H_2^3 \\ H_3^1 & H_3^2 & H_3^3 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ 1 \end{pmatrix}$$

- One value of H can be set arbitrarily, then there are 8 unknowns.
- Each (x, x') pair provides 2 equations then at least 4 pairs are needed.
- However, since many matches are unreliable in practice, much more pairs are actually needed!

How to get rid of outliers in homography estimation?

The RANSAC (RANdom SAmple Consensus) algorithm is an iterative stochastic method for estimating a model from a large collection of observations containing outliers:

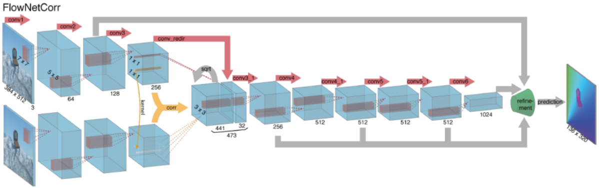
$e_{\min} = \infty$
For $k = 1$ to $K_{\max}$:
  Randomly choose N pairs among A to form $B \subset A$
  $\hat{H} \leftarrow$ Estimate_Homography(B)
  $n_{inliers} \leftarrow 0$
  For each pair j such that $(x_j, y_j) \notin B$:
    If $||y_j - \hat{H}x_j|| < \varepsilon$:
      $n_{inliers} \leftarrow n_{inliers} + 1$
      $B \leftarrow B \cup \{(x_j, y_j)\}$
  If $n_{inliers} < T_{inliers}$:
    $e \leftarrow$ Total_error($\hat{H}, B$)
    If $e < e_{\min}$:
      $H_{best} = \hat{H}$
      $e_{\min} = e$

- $A = \{(x_i, y_i)\}_i$ the set of matched pairs
- N the number of pairs used to estimate the homography ($N \geq 4$)
- $K_{\max}$ the number of iterations
- ε a distance threshold
- $T_{inliers}$ the minimal number of inliers
- H_{best} the best homography

Conclusion: Benchmarks and current trends

From the 80's to now, Optical Flow estimation has remained a very active research domain.

- Thousands of publications: few principles, many different recipes...
- Several popular benchmark datasets with ground truth:
  - `http://vision.middlebury.edu/flow/` (Indoor)
  - `http://www.cvlibs.net/datasets/kitti/eval_scene_flow.php?benchmark=flow` (Urban outdoor)
- End-to-end deep learning methods are emerging massively, with outstanding performance and increasing computational efficiency.



CNN based OF estimation: Flownet **[Fischer 15]**

Feature Matching and Optical Flow - CONCLUSION

What have we learned so far

- Interest of the Optical Flow
- Difficulties and Constraints
- Local Sparse Baseline Method: Lucas and Kanade
- Global Dense Baseline Method: Horn and Schunck
- Multiscale Approaches
- Optical Flow and RANSAC for Homography estimation
- Emerging Trends

Upcoming Lectures

- 3d vision and Navigation.
- Object and scene modelling